UNIVERSIDADE DE LISBOA
Faculdade de Ciências
Departamento de Engenharia Geográfica, Geofísica e Energia

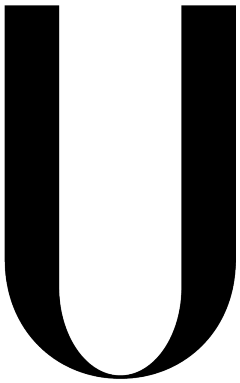
Criação de plataforma de *geocoding* baseada em serviços Google Maps

Stephane Goldstein

Projeto
Mestrado em Sistema de Informação Geográfica
Tecnologias e Aplicações

2014

UNIVERSIDADE DE LISBOA
Faculdade de Ciências
Departamento de Engenharia Geográfica, Geofísica e Energia



LISBOA

UNIVERSIDADE
DE LISBOA

Criação de plataforma de *geocoding* baseada em serviços Google Maps

Stephane Goldstein

Projeto
Mestrado em Sistema de Informação Geográfica
Tecnologias e Aplicações

Orientadores:
Cristina Maria Sousa Catita
Sandro Gonçalo da Fonseca Batista

2014

SUMÁRIO

Este projeto foi realizado no âmbito de um estágio na consultora de negócios *Focus BC*. De acordo com as necessidades da empresa, a proposta constitui a criação de uma plataforma de georreferenciação de moradas postais em lote, utilizando o serviço de *geocoding* da *Google Maps*: a *Geocoding API*. A aplicação foi escrita na linguagem de programação *Python*, e conta com etapas de pré e pós processamento dos dados que possibilitaram aumentar a qualidade dos resultados oferecidos pelo serviço.

Métricas de avaliação dos dados originais foram criadas (métricas de entrada), permitindo ajustar as expectativas quanto aos possíveis resultados, assim como estimar o esforço necessário para obtenção da precisão desejada na georreferenciação.

Em seguida os dados passam por um módulo de correção e normalização de moradas, ficando prontos para serem enviados ao serviço. Sucessivos pedidos de *geocoding* para um mesmo registo são eventualmente feitos, alterando a estrutura da morada de acordo com um fluxo de controle específico.

Como as quatro classes de precisão dos resultados atribuída pela *Geocoding API* não possuem um nível de detalhe suficiente no contexto dos projetos desenvolvidos pela *Focus BC*, foi implementado um método que classifica os resultados em sete categorias.

A versão final da aplicação georreferenciou 31915 moradas provenientes de quatro fontes distintas. Uma precisão considerada aceitável foi atingida para 81.8% dos registos. A normalização e o sistema de múltiplos pedidos lograram em conjunto um aumento de 9.53 pontos percentuais na proporção de registos referenciados com precisão considerada alta (55% do total). Apesar da normalização, os resultados são influenciados pela qualidade dos dados originais, como foi possível evidenciar através das métricas de entrada.

Uma verificação manual da precisão de 2% dos registos permitiu avaliar a confiança da classificação dos resultados, indicando uma precisão global de 91.72 %.

No âmbito deste estudo identificaram-se algumas melhorias possíveis de serem implementadas nos processos de normalização e na atribuição de precisão. Justifica-se a necessidade de futuramente desenvolver-se uma interface web para facilitar o uso da aplicação, assim como sua adaptação a moradas e organizações territoriais de outros países.

Palavras chave: geocoding, georreferenciação, Google Maps API

ABSTRACT

The project hereby presented was realized in the context of an internship at the business consulting company *Focus BC*. Seconding the company needs, it consists in the development of a batch geocoding platform, based on the geocoding service provided by *Google Maps*: the *Geocoding API*. The application was written in the *Python* programming language and includes pre and post data processing operations that allowed to obtain results of better quality than those provided by the geocoding service alone.

Quality metrics for the original data were developed, making possible to adjust the expectations in terms of possible achievable results during the geocoding process. Those metrics also help to estimate the amount of effort needed in order to geocode a specific set of data with a determined precision.

Following this first evaluation, data undergo a correction and normalization process, making it ready to be sent to the geocoding service. By the means of an established control flow, successive geocoding requests for the same record are eventually made, but using different postal address structures.

The four precision classes provided by the *Geocoding API* to classify the results do not have the necessary level of detail needed in the context of the projects developed by *Focus BC*, so a method that classifies the results within seven precision categories was implemented.

The application's final version was used on the referencing of 31915 records from four different sources. A precision considered acceptable was achieved for 81.8% of those records. The normalization and multiple geocoding system altogether enabled a 9.53 percentage points increase in the proportion of addresses referenced with a precision considered high. Despite the normalization process, results greatly depend on the original data quality, as it was possible to confirm by the means of the original data metrics.

A manual verification of the precision on 2% of the records allowed to evaluate the confidence of the results classification method. A global precision of 91.72% was determined.

In the context of this study possibilities of improvements were identified in the normalization and results classification steps. There is also justification for future works to be considered, adapting the application to other countries addresses and territorial organization, as well as the creation of a web interface in order to improve its usability.

Keywords: geocoding, georeferencing, Google Maps API

Índice

1. INTRODUÇÃO

O projeto descrito neste documento foi realizado no âmbito de um estágio na *Focus BC* (Focus BC 2014), empresa de consultoria que se posiciona no nível de decisão do negócio de seus clientes, oferecendo serviços nas áreas de Consultoria de Gestão, Inteligência Geográfica, Estudos e Inteligência de Mercado, Comunicação e Colaboração Empresarial, e Redes Sociais e Colaboração.

A materialização dos serviços de consultoria é feita através da conceção, desenvolvimento e implementação de soluções que provoquem impactos organizacionais pela criação de valor no negócio de seus clientes, medidos em termos do aumento da eficácia do negócio, níveis mais elevados de eficiência operacional, maior capacidade de relacionamento com o cliente e melhoria dos processos de tomada de decisão.

O projeto surgiu da necessidade cada vez mais comum que se tem de "colocar moradas no mapa". Atualmente, quando se quer saber onde se situa um local, acede-se à algum dos serviço existentes de mapas na web, como o *Google Maps* e faz-se uma pesquisa, logo de seguida o local procurado aparece em um mapa. Este processo é denominado de *geocoding* e para o utilizador final que procura apenas uma ou duas moradas é hoje em dia algo comum e fácil.

A situação porém é completamente diferente no caso do *geocoding* de bases de dados maiores, às vezes com centenas de milhares de registos. Como a maioria das empresas, os clientes da *Focus BC* possuem bases de dados com as moradas de seus clientes, vendedores, fornecedores, etc. A melhor forma de se explorar estes dados no âmbito dos serviços de consultoria em Inteligência Geográfica oferecidos pela *Focus BC*, é através do seu *geocoding*. O desenvolvimento de um sistema automatizado que execute esta tarefa torna-se então necessário.

A *Focus BC* é parceira da *Google* para o mercado EMEA (Europa, Médio Oriente e África), oferecendo toda a cadeia de valor dos produtos *Google Maps for Work*. Esta parceria resulta do reconhecimento das competências técnicas da empresa nas vertentes de P*erformance Management*, *Location Intelligence* e Integração de Sistemas.

Os produtos e serviços oferecidos pela *Google Maps for Work* incluem:

- *Google Maps Engine*: plataforma web para armazenamento e edição de dados espaciais, criação e partilha de mapas na web (Google 2014e)

- *Google Maps Coordinate*: ferramenta de gestão de equipas de trabalho em campo, apoiada em informação geográfica (Google 2014d)

- *Google Earth Pro*: software de visualização e edição de dados geográficos sobre um globo interativo em 3D (Google 2014a)

- *Google Maps API*: conjunto de serviços oferecidos pelo *Google Maps*, como cálculo de rotas, matriz de distâncias, altimetria, imagens aéreas, vistas de rua e nomeadamente, *geocoding* de moradas.

O objetivo do projeto foi portanto criar uma aplicação que georreferencie grandes quantidades de dados, utilizando o serviço de *geocoding* da *Google Maps*. O desenvolvimento da aplicação iniciou-se da forma mais simples possível: fazendo os pedidos e recuperando os resultados. Com o tempo, diversas melhorias foram implementadas e a aplicação tornou-se mais complexa. Foram criadas métricas que avaliam a qualidade dos dados a serem processados, permitindo fazer uma previsão aproximada dos resultados. Foram criados métodos para a otimização dos resultados, através da normalização e correção de erros nas moradas e da criação de um sistema que faz diversas tentativas de *geocoding* para uma mesma morada, alterando alguns elementos da mesma. Por último foi criado um sistema de atribuição de precisão aos resultados retornados.

Este documento numa primeira parte define alguns conceitos de representação da informação espacial qualitativa e quantitativa, assim como aborda um pouco da teoria existente sobre a georreferenciação de moradas. Na segunda parte do trabalho a aplicação é descrita tanto globalmente por uma visão de alto nível, como detalhadamente em cada uma das suas etapas. Na terceira parte, os resultados obtidos são avaliados e discutidos. Por último algumas considerações sobre a aplicação e possibilidades futuras são explicitadas.

2. CONTEXTO

2.1. Representações da informação espacial e *geocoding*

A busca de um método preciso para comunicação de localizações iniciou-se cedo na historia da geografia. Em aproximadamente 150 A.C, o grego Hiparco dividiu a terra em meridianos e paralelos, definindo assim o primeiro método para descrever um ponto no globo: as latitudes e longitudes (Cote 2014).

Passados dois mil anos, continuamos a utilizar latitudes e longitudes, assim como diversos outros sistemas de coordenadas, para descrever localizações. Os sistemas de coordenadas com que trabalhamos são uma representação quantitativa da informação espacial. Isto significa que utilizam valores numéricos para indicar de forma unívoca um local na superfície da Terra (Felice 2012).

Os Sistemas de Informação Geográfica (SIG) utilizam dados quantitativos para armazenar geometrias que representam a informação geográfica. Por serem dados numéricos, estas geometrias podem ser analisadas através de algoritmos estatísticos e de geometria computacional, permitindo a execução de operações complexas como por exemplo interpolação de superfícies e cálculos de rotas (Felice 2012).

A representação quantitativa da informação espacial tem assim vantagens em termos de precisão e aplicações computacionais, porém somente consegue representar alguns dos aspetos do espaço geográfico. A conceptualização de alto nível existente na forma como o ser humano apreende o que existe ao seu entorno não é considerada neste método (Bittner & Stell 1999; Felice 2012).

Diferentemente dos SIG, as pessoas utilizam a linguagem natural para construir mapas mentais do espaço geográfico e transmitir esta informação (Yao & Jiang 2005). Isto constitui um método qualitativo para representação da informação geográfica. As localizações qualitativas são utilizadas para descrever locais e relações espaciais, e mesmo sem especificar pontos precisamente referenciados, estas descrições são facilmente interpretadas pelos humanos (Yao & Jiang 2005; Felice 2012).

Informações do tipo “após a bomba de gasolina, na segunda rua à esquerda” ou “na parte de cima do parque Eduardo VII” são compreendidas sem problemas por pessoas num determinado contexto. Os serviços de emergência são o principal exemplo de disto, pois

localizam os pontos de incidentes utilizando principalmente informação qualitativa. (Felice 2012).

As localizações qualitativas podem descrever informação a várias escalas e níveis de detalhe (Yao & Jiang 2005). Sabemos que o "Algarve" é uma região que se encontra em "Portugal". Existem fronteiras bem definidas para ambos os elementos, assim como uma relação espacial entre eles. O "centro de Lisboa" em contrapartida, se encontra na cidade de Lisboa, mas não tem limites bem definidos. Em termos do nível de detalhe, podemos determinar uma zona como "a Faculdade de Ciências" ou de forma mais específica, "a cafetaria do edifício C8 da FCUL" .

A forma mais comum que se utiliza para comunicar localizações qualitativas é com base na toponímia, ou seja, a atribuição de nomes a elementos geográficos. Estes elementos podem ser aglomerados populacionais, divisões administrativas (freguesias, concelhos, países), elementos naturais (rios, montanhas, lagos), infraestruturas (vias de comunicação, aeroportos, hospitais, edifícios, quarteirões) ou ainda elementos sem limites definidos porém compreendidos em um contexto local (e.g. uma zona de pasto ou local de pesca)(United Nations Group of Experts on Geographical Names 2006).

Dicionários de topónimos são chamados *gazeteers* (Hill 2006). Em geral contêm uma descrição do local, informações de eventual interesse, assim como suas coordenadas ou representação em um mapa. Os primeiros registos da existência de tais dicionários datam do século VI, porém somente se popularizaram e tomaram uma forma mais estruturada a partir do século XVIII, no contexto dos avanços científicos ocorridos no domínio da geografia (Hill 2006).

Os *gazetteers* comummente estão acompanhados de um mapa, onde, a partir do índice pode-se localizar o elemento desejado. Na Figura 1 pode-se ver um *gazetteer* datado de 1905, do atual estado de Oklahoma nos Estados Unidos, assim como a correspondência que é feita entre topónimos e posições coordenadas no mapa (assinalados a vermelho para a localidade de Kullituklo). Os *gazetteers* são um método de conversão da informação geográfica qualitativa para a quantitativa. Este processo de georreferenciação da informação também é designado de *geocoding* (Goldberg 2008).

Para facilitar a comunicação verbal da informação espacial, atribuem-se nomes à maioria dos elementos do nosso entorno. Porem nos ambientes urbanos densamente povoados em que vivemos hoje, os topónimos não constituem informação suficiente para encontrarmos um edifício ou estabelecimento determinado (Hill 2006). Encontrar por exemplo o Cinema São

Jorge, somente sabendo que ele se situa “na Avenida da Liberdade em Lisboa” não é uma tarefa fácil, pois ao longo desta avenida existem centenas de edifícios.

Figura 1: Gazetteer do atual estado de Oklahoma (EUA) datado de 1905 (Geographical Publishing Co. 1905)

Para poder identificar de forma unívoca os locais de interesse, utiliza-se um conjunto de dados específicos e formatados segundo um padrão determinado. A este conjunto de dados atribui-se o nome de “morada” ou “morada postal” (MP), como será designada adiante. Os dados que a compõem, assim como sua estrutura, variam consoante o país (Davis Jr. & Fonseca 2007). Utiliza-se em geral uma mistura de topónimos e informação numérica para construí-la (Li & Zhang 2010). Em Portugal o modelo de MP mais utilizado é composto pelos topónimos do nome do concelho e/ou da localidade, e o nome da via. Os valores numéricos são o número da porta na via em questão, eventuais complementos (piso, lado, letra, etc) e o código postal (Universal Postal Union 2013). A ordem dos elementos também varia consoante o país. Em Portugal e em França, os elementos que compõem a MP são os mesmos. Porem em Portugal o

número de porta se situa após o nome da via, enquanto que em França se situa antes. As MPs em ambos países ficam na forma dos exemplos:

- Avenida da Liberdade 17, 1250-141 Lisboa, Portugal
- 17 Rue de la Liberté, 75019 Paris, France

As MPs são a principal informação geográfica armazenada em muitos sistemas de informação, tanto públicos como privados. Alguns exemplos são bases de dados de ocorrência de crimes, acidentes viários, doenças, irregularidades sanitárias e infraestruturas. No setor privado podemos destacar as bases de dados de clientes, potenciais clientes, estabelecimentos próprios, estabelecimentos da concorrência e fornecedores. A exploração desta informação pelas ferramentas de análise oferecidas pelos SIG somente é possível caso se faça seu *geocoding*, ou seja, a transformação de MPs em pontos coordenados.

Qualquer aplicação de *geocoding* de MPs, em sua conceptualização de mais alto nível, pode ser separada em 5 elementos, conforme ilustrados na Figura 2.

A primeira etapa consiste no tratamento da MP, corrigindo eventuais erros e adequando-a à um formato que seja compatível com as demais etapas do processo. A análise sintática (*parsing*) permite separar as diferentes componentes que constituem a MP, como o nome da via, o número de porta, o concelho, e o código postal. A normalização é importante para remover elementos desnecessários. Informações como o andar ou o lado de um apartamento, além de não serem necessárias para o *geocoding*, podem prejudicar na pesquisa do local.

Na maior parte dos casos, as MPs são obtidas por inserção manual em bases de dados. Isto significa que são encontradas diferentes formas de abreviaturas, separação de campos com diferentes caracteres ou ainda erros de digitação. Esta primeira etapa de processamento deve contemplar todas estas questões. Se por um lado o nome do concelho, localidade e código postal não supõem um problema complexo, o nome da via, com seu número e demais complementos apresentam mais dificuldades. Este sub-conjunto de dados, que compõem a primeira parte da MP, será de agora em diante designado por “morada”. A Tabela 1 mostra alguns exemplos de moradas inseridas em bases de dados, e como estas ficariam após uma normalização, com substituição de abreviaturas e eliminação de elementos desnecessários para o *geocoding*.

Os elementos 2, e 4 no diagrama da Figura 2 constituem em conjunto o "algoritmo de *geocoding*". Como os objetivo propostos no âmbito deste projeto não englobam o processo de *geocoding* em si, que fica a cargo da *Google Maps*, estas etapas não serão descritas em detalhe. Basicamente, o algoritmo de pesquisa (elemento 2) recebe os dados normalizados e busca uma correspondência entre a MP e a base de dados espacial (elemento 3). O modelo de dados da mesma deve se adequar tanto às informações disponíveis como à forma como as MPs do local em questão são estruturadas.

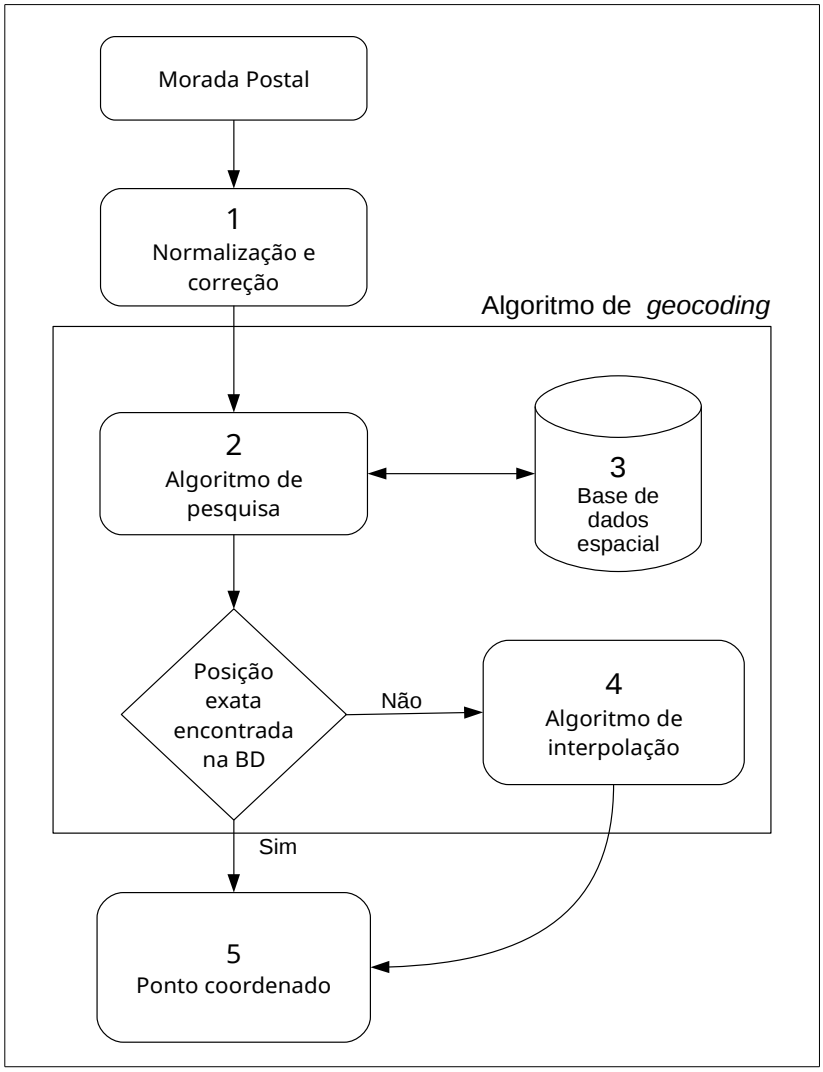


Figura 2: Etapas do processo de geocoding (adaptado de Goldberg 2008)

Tabela 1: Exemplos de moradas mal formatadas e formato normalizado

Morada recebida	Morada normalizada
R DA RIBEIRA N 23 BLOCO B R/C B -PEDROGOS	RUA DA RIBEIRA 23
RUA.PARTICULAR-A AV DO BRASIL-2	AVENIDA DO BRASIL 2
R MANUEL TEIXEIRA GOMES N.51 3 DTO.	RUA MANUEL TEIXEIRA GOMES 51
URB QUINTA DA OLIVEIRA BAIRRO S MIGUEL LOTE 4 2 DIR	URBANIZAÇÃO QUINTA DA OLIVEIRA 4
R 25 ABRIL , AVEIRAS BX°	RUA 25 DE ABRIL
R JOÃO BARROS LT.480 3°FRT	RUA JÕAO BARROS 480
R. JOSE PEDRO DA SILVABL 1 LJ 11 A	RUA JOSÉ PEDRO DA SILVA 1

Enquanto que em Portugal as MPs são construídas com base nos topónimos das vias de comunicação, na China elas são construídas com base nos topónimos dos quarteirões. Considerando estas diferenças, o modelo de dados pode ser constituído por distintas primitivas geográficas, como linhas representando vias de comunicação, áreas representando quarteirões e lotes, ou ainda pontos com uma ou mais moradas específicas associadas (Goldberg 2008).

O 4° elemento no diagrama consiste no processo de interpolar uma posição para a MP caso a base de dados espacial não contenha informação exata sobre a mesma. Para uma via, por exemplo, caso conste apenas a posição do número inicial e final da mesma, um edifício com número intermédio tem nesta etapa a sua posição interpolada ao longo da via.

Todas os elementos do diagrama influenciam na qualidade do *geocoding* e o resultado final depende da correta normalização dos dados, das capacidades do algoritmo de *geocoding* e da completude da base de dados espacial. Na 5ª e última etapa as coordenadas do local são retornadas, acompanhadas eventualmente de metadados sobre o processo, como a morada normalizada ou a precisão de resultado.

2.2. Opções de ferramenta para *geocoding* em lote

Quando existe a necessidade de se fazer o *geocoding* de muitos dados, duas abordagens são possíveis. Uma delas é administrar todo o processo localmente, incluindo a base de dados espacial, o algoritmo de *geocoding* e a exportação dos resultados. A base de dados pode ser pública, como o *TIGER* (U.S. Census Bureau 2014), que contém informação geográfica sobre todas as ruas dos Estados Unidos, pode ser proveniente de *crowdsourcing*, como o projeto *OpenStreetMap* (OpenStreetMap Contributors 2014), ou ainda adquirida junto a empresas como a *Navteq* (Navteq Maps 2014) ou *TomTom* (TomTom 2014).

Em seguida deve-se adequar a base de dados de forma a que ela funcione com alguma aplicação que oferece algoritmos de *geocoding*, como o *Oracle Spatial* (Oracle 2014), o *ArcGIS for Desktop* (ESRI 2014) ou o *PostGIS* (PostGIS Project Steering Committee 2014).

A outra abordagem possível é a utilização de serviços web. A principal vantagem deste método está no facto de delegar-se a manutenção da base de dados espacial. Com o passar do tempo, a disponibilidade e qualidade dos dados espaciais aumenta e o espaço geográfico é alterado. A utilização de um serviço de qualidade garante que os dados utilizados no processo estarão sempre atualizados.

A qualidade dos algoritmos de *geocoding* existentes também pode vir a ser melhorada e caso sejam implementadas melhorias no serviço escolhido, isto não se traduz no licenciamento e instalação de um software mais recente, como seria necessário na primeira abordagem. A lista de empresas que oferecem serviços web de *geocoding* é longa, mas podemos destacar algumas como a *Google Maps* (Google 2014b), a *Bing* (Microsoft 2014) e a *GeoCoder Pro* (incratec GmbH 2014).

Sendo a *Focus BC* parceira da *Google Maps*, a escolha do serviço para o desenvolvimento da aplicação fica clara e não está no escopo deste trabalho comparar serviços de *geocoding*. Utilizando o serviço de *geocoding* da *Google*, temos uma solução completa de georreferenciação, que abarca as 5 etapas que foram descritas. Não sendo possível intervir nas etapas 2, 3, 4 e 5 do processo, a forma de otimizar os resultados do serviço se encontra no pré-processamento dos dados (o que corresponde a etapa 1) e no pós processamento dos dados devolvidos na etapa 5. Apesar do serviço também executar a normalização das MPs, quanto mais isentos de erros e abreviaturas estiverem os dados de entrada, maior serão as possibilidades do serviço encontrar o local desejado.

3. MÉTODOS

3.1. Escolha das ferramentas de trabalho

3.1.1. Escolha da API para os pedidos de *geocoding*

Uma vez definido que a aplicação trabalhará com o serviço de *geocoding* da *Google*, foi necessário avaliar quais as ferramentas programáticas mais adequadas para sua concretização. Para poder aceder o serviço de *geocoding* oferecido pela *Google Maps*, existem duas Interfaces de Programação de Aplicações (*API – Application Programming Interface)* disponíveis: uma na linguagem *JavaScript (ECMA International 2011)*, e outra baseada em serviços do tipo *HTTP REST (Fielding 2000)*.

A *Google Maps JavaScript API* (Google 2014f) foi implementada de forma a facilitar a introdução dos serviços *Google Maps* em páginas *HTML* dinâmicas. É uma *API* destinada à criação aplicações na web que são executadas pelo navegador do cliente.

A *Google Maps API Web Services* (Google 2014c) é um conjunto de *APIs HTTP REST* que dão acesso aos diversos serviços que fazem parte do *Google Maps*. Os serviços *HTTP REST* funcionam basicamente através de um pedido *HTTP* que é feito pelo cliente, na forma de uma *URL*, construída de maneira a incluir as informações relacionadas a este pedido.

O servidor recebe esta *URL*, e devolve ao cliente uma resposta que pode vir em diversos formatos, como um ficheiro ou uma página *HTML*. *N*o caso dos dos serviços *HTTP REST* do *Google Maps*, pode-se escolher entre um objeto *JSON* (ECMA International 2013) ou *XML (World Wide Web Consortium 2008)*. Estes objetos são constituídos por uma serie de pares atributo / valor, contendo as informações solicitadas.

Por exemplo, caso se pretenda fazer um pedido a *Google Elevation API* (Google 2014g) para saber a que altitude se encontra a Faculdade de Ciências (latitude 38.757 e longitude -9.156), a *URL* do pedido deve ser construída agregando:

- A URL do serviço: `http://maps.googleapis.com/maps/api/elevation/`
- O tipo de retorno que se deseja (*JSON* ou *XML*)
- O atributo `location`, com a latitude e longitude do local que se deseja obter a altitude

A URL final do pedido que deve ser feito é então:

```
http://maps.googleapis.com/maps/api/elevation/json?locations=38.757,-9.156
```

Que o servidor reponde com o seguinte texto do tipo JSON:

```
{
 "results" :
       [{ "elevation" : 80.44493865966797,
          "location" : {"lat" : 38.757, "lng" : -9.156},
          "resolution" : 76.35161590576172
     }],
 "status" : "OK"
}
```

A forma como os pedidos devem ser feitos e como a resposta está estruturada são especificas a cada *API* e em geral constam na documentação da mesma. A vantagem dos serviços *HTTP REST* com relação a *API* em *JavaScript*, é que os serviços *REST* não dependem de nenhum tipo de linguagem especifica, podendo ser utilizados com qualquer ferramenta que faça pedidos *HTTP* e recupere a resposta. Optou-se portanto utilizar esta via de acesso ao serviço de *geocoding*, permitindo uma livre escolha da linguagem mais eficaz e prática para se escrever a aplicação proposta.

3.1.2. Escolha da linguagem de programação

Como foi visto, o único pré-requisito na escolha da linguagem de desenvolvimento até o momento, é a capacidade de se fazer pedidos *HTTP* e recuperar o resultado. Inúmeras linguagens possuem tal funcionalidade. Porém, considerando que a aplicação irá trabalhar com grandes quantidades de dados, seu desenvolvimento ver-se-ia muito facilitado sendo feito com uma linguagem de alto nível, que tenha gestão de memória incorporada, e também que seja indicada para o processamento de dados.

Ao longo do desenvolvimento de uma aplicação, sempre existe a possibilidade de alteração dos requisitos e métodos utilizados. Optar por uma linguagem que seja facilmente extensível com bibliotecas adicionais é assim uma mais-valia.

Entre as linguagens avaliadas no âmbito deste projeto (*R*, *Matlab*, *Java*, *SQL*, *JavaScript, PHP* e *Python*), as que melhor cumprem os requisitos estabelecidos são *Java (Gosling & McGilton 1996)* e *Python* (Python Software Foundation 2014a). Optou-se pelo *Python* essencialmente pela existência do módulo para computação cientifica *NumPy* (Numpy developers 2014), que permite trabalhar com grandes quantidades de dados de forma rápida e relativamente fácil.

Outro fator que influenciou nesta escolha foi a existência de um módulo de código aberto para *geocoding* com serviços *Google*: o *pygeocoder* (Yu 2014). Este módulo foi alterado e melhorado no âmbito deste projeto, e as modificações foram revertidas ao projeto original. O módulo resultante foi incorporado como uma das peças da aplicação.

3.2. Componentes da aplicação

A aplicação está estruturada em peças, de acordo com as várias etapas de processamento as quais os dados são submetidos conforme listado abaixo e organizados de acordo com a Figura 3:

1. Importação dos dados originais
2. Avaliação da qualidade dos dados originais
3. Normalização e correção das moradas
4. Pedido de *geocoding* (diversos tipos de *geocoding* possíveis)
5. Verificação da precisão do resultado
6. Repetição do *geocoding* quando necessário
7. Métricas de precisão final
8. Exportação dos resultados finais

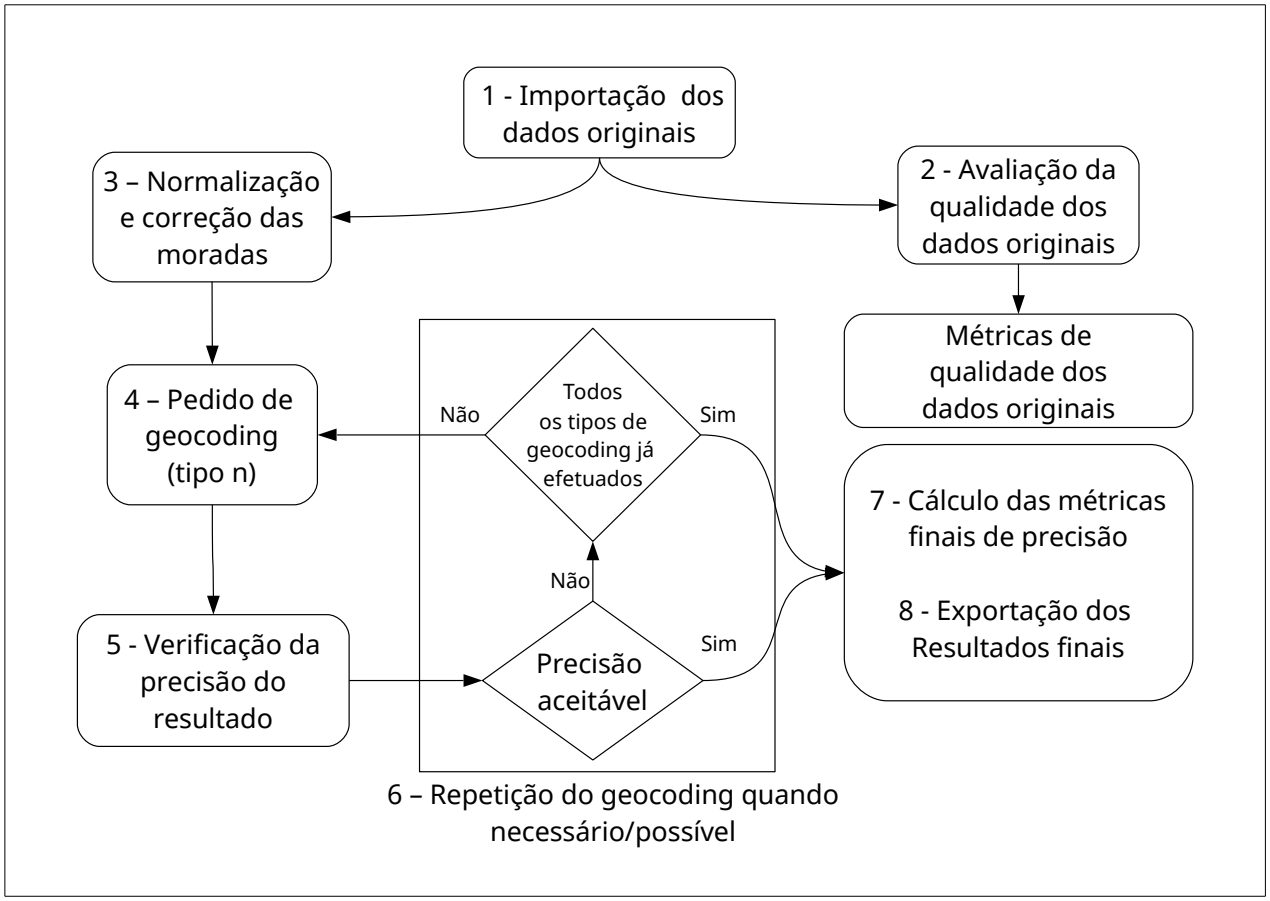


Figura 3: Fluxograma de processamento dos dados

3.3. Base de dados de apoio

Em algumas das etapas da aplicação, nomeadamente nas etapas 2 (Avaliação da qualidade dos dados originais) e 4 (pedido de *geocoding*), recorreu-se a fontes de dados externas para auxiliar no processamento dos dados. Os dados utilizados foram:

- O "Ficheiros de Códigos Postais" (CTT - Correios de Portugal SA 2014) fornecido pelos *CTT*, que contém a lista de todos os códigos postais de 7 dígitos (CP7) de Portugal, assim como a localidade, concelho e distrito em que se encontram.
- Mapa vetorial dos limites dos códigos postais de 4 dígitos (CP4) (CTT - Correios de Portugal SA 2014), também obtido junto aos *CTT*. A partir deste mapa, extraíram-se as coordenadas dos limites da área de abrangência (*bounding-box*) de cada CP4.

Estes dados foram processados, removendo-se atributos desnecessários e formatando-os de forma a permitir a sua interação com a aplicação. Devido ao fato dos dados serem estáticos e da aplicação não fazer alterações sobre os mesmos, optou-se por não armazená-los em uma base de dados relacional, mas sim carregá-los na memória *RAM* do computador, durante a inicialização da aplicação, na forma de tabelas *NumPy* importadas a partir de ficheiros de texto.

Apesar desta escolha se traduzir em um ligeiro aumento no tempo de arranque da aplicação, a velocidade de processamento dos dados durante a execução do programa vê-se muito aumentada, devido ao acesso aos dados estar sendo feito na memória volátil, de velocidade muito superior a do disco rígido. O diagrama da Figura 4 ilustra o modelo de dados criado nas tabelas *NumPy*.

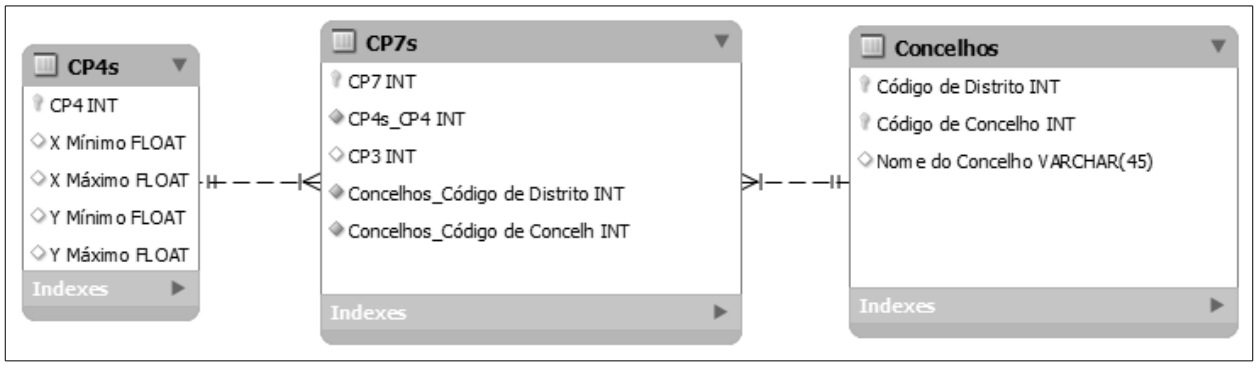


Figura 4: Modelo de dados da base de dados de apoio

3.4. Importação dos dados

Para possibilitar a receção de dados de diversas fontes, foi criado um ficheiro modelo, na forma de uma tabela tipo *Comma Separated Value* (CSV) (The Internet Engineering Task Force 2005) onde as moradas devem ser inseridas. As colunas da tabela consideram os diversos elementos que podem compor uma MP e tomam a seguinte forma:

- ID : Identificador único do registo
- Nome: nome do estabelecimento ou local, caso exista
- Morada: nome da via / bairro / lugar, número de porta, complemento (andar, lado, etc)
- Código Postal: CP7 ou CP4, conforme disponibilidade
- Localidade
- Concelho
- Distrito
- País

Todos os campos com exceção do ID são de preenchimento opcional. Obviamente que quanto mais campos à branco mais difícil se torna a correta georreferenciação do registo. Algumas aplicações ou bases de dados de que trabalham com MPs separam os campos de forma mais detalhada, subdividindo a morada em mais campos, como "via" e "número de porta e "complemento", ou ainda "tipo de via" (Avenida, Rua,...), "nome da via", "número de porta" e "complemento". Existem ainda bases de dados que tem o código postal separado em dois campos, um com o CP4 e outro com os últimos três dígitos do CP7 (CP3).

Optou-se por não subdividir os campos tão detalhadamente, já que a maior parte dos dados recebidos pelos clientes da *Focus BC* não contavam com tanta desagregação da informação. Em geral discriminavam apenas os campos "Morada", "Código Postal" e "Concelho" ou "Localidade". As dificuldades de se dividir o campo "Morada" nas suas diversas componentes são muitas, principalmente considerando o formato totalmente desnormalizado em que os dados recebidos geralmente se encontram. Também não é necessário que este processo de *parsing* seja executado pela aplicação, pois fica a a cargo do serviço web desagregar as componentes da morada para buscar correspondências na base de dados espacial.

3.5. Avaliação da qualidade dos dados originais

Ao receber os dados a serem georreferenciados, é importante ter alguma medida da sua qualidade inicial. Isso permite identificar motivos de possíveis problemas na referenciação, assim como ajustar as expectativas quanto a qualidade dos resultados que podem ser obtidos.

Com este objetivo em mente, foram criadas três medidas da qualidade dos dados de entrada, utilizando os dados de apoio previamente mencionados. Para cada registo a ser georreferenciado, são verificados os seguintes elementos:

- Verificação do CP7 / CP4

Com a lista de todos os CP7, verifica-se se o CP7 do registo existe na base de dados dos *CTT*. Muitas vezes o CP7 do registo não existe ou corresponde a um apartado. Os apartados são recetáculos postais localizados nas estações de correio, e portanto não auxiliam na georreferenciação da MP. Existem mais de 30 mil CP7 em Portugal, sendo esta uma das informações de maior importância para se localizar uma MP corretamente. A aplicação verifica para cada registo a existência do CP4 indicado. Caso exista, verifica a existência do CP3.

- Verificação de existência do Concelho

Outra verificação realizada é a existência do concelho do registo na lista dos concelhos Portugueses. Devido a erros de digitação ou acentuação, existe a possibilidade que a forma como o nome do concelho foi escrito em um registo não corresponda a sua nomenclatura oficial. Verificar a existência de um concelho fazendo uma comparação estrita entre estas duas sequências de caracteres (*strings*) iria dar lugar a muitos falsos negativos (por exemplo, "V. N. DE GAIA" é textualmente diferente de "VILA NOVA DE GAIA", porém trata-se do mesmo concelho). Para ultrapassar estas diferenças, a comparação entre concelhos é feita utilizando o método da Distancia de Levensthein (Levensthein 1966). Este método é também utilizado mais a frente, no processo de atribuição da precisão ao *geocoding*, pelo que será explicado mais em detalhe nesta parte, onde toma mais importância. De forma simplificada, pode-se dizer que utilizando este algoritmo para comparar duas *strings*, conseguimos uma medida de similaridade entre ambas. A aplicação considera que o concelho do registo existe quando haja na lista oficial de concelhos algum cujo o nome tenha mais de 65% de similaridade.

- Correspondência concelho / código postal

A ultima métrica de qualidade avaliada é se o concelho e código postal fornecidos coincidem. Procura-se na tabela dos *CTT* o código postal do registo e em qual concelho este se localiza. Em seguida compara-se este concelho com o concelho do registo. Uma similaridade de mais de 65% entre ambos indica que o concelho do registo corresponde ao seu código postal.

Na maior parte dos casos, os limites de um CP4 estão contidos dentro de um único concelho. Nestes casos é possível determinar a correspondência mesmo que o registo só contenha o CP4 (ou contenha o CP7, mas o CP3 esteja incorreto). Alguns CP4 tem a sua área de

abrangência sobre parte de dois concelhos, portanto é necessário saber o CP7 para encontrar o concelho correto na lista dos *CTT*. A aplicação contempla ambos os casos, e determina sempre que possível a relação. Com base nos 3 indicadores criados, as métricas dos dados de entrada ficam na forma do exemplo da Tabela 2.

Tabela 2: Exemplo de métricas da qualidade dos dados de entrada

Indicador	Nº. de registros
Registos totais	1854
Registos com CP inexistente ou correspondente a um apartado	0
Registos com um CP4 existente	1854
Registos com um CP7 existente	1678
Registos com um concelho existente	845
Registos com um concelho que corresponde ao seu código postal	801

3.6. Correção e normalização dos dados

Em geral, nas bases de dados recebidas os campos código postal e concelho ou localidade já vêm discriminados, e não necessitam nenhum processamento complexo. O principal desafio da normalização dos dados se encontra no campo “Morada”. Como foi visto anteriormente, este campo contêm muitos elementos que são não só desnecessários como prejudiciais para a georreferenciação. Atributos como andar, porta, lote, interseção e bairro devem ser removidos. Podem estar escritos de diversas formas, serem compostos por números ou letras e estar separados por diferentes caracteres. Existem ainda moradas que fogem do padrão mais comum que foi descrito, como urbanizações, quintas, bairros, zonas comerciais e industriais.

Considerando que a composição e formato das moradas são em geral resultado da sua inserção manual nas bases de dados, as variações possíveis são ilimitadas e a criação de um método automatizado de normalização que consiga corrigir todos os erros é uma tarefa inexequível. Entretanto podemos contemplar os casos mais comuns tentando resolver o máximo de discrepâncias possíveis. A técnica mais indicada para este tipo de operações consiste na utilização de expressões regulares (REGEX) (Friedl 2002).

O REGEX é uma forma flexível de identificar sequências e padrões de caracteres. Pode ser visto como uma ferramenta "localizar e substituir", porém mais avançada e com uma sintaxe que permite encontrar vários padrões de caracteres, simples ou complexos. Esta sintaxe é curta

e concisa, tornando a legibilidade do código um desafio no caso de expressões mais complexas. Trabalhando em *Python*, utilizou-se o modulo de REGEX `re` (Python Software Foundation 2014b).

Ilustrando a flexibilidade deste método, a seguinte expressão:

```
^AL(AM?)?(\.? |\.)
```

permite localizar as seguintes abreviaturas possíveis (em sublinhado) da palavra "Alameda" para a morada "Alameda dos Oceanos" (tanto em caixa alta como caixa baixa):

AL_DOS OCEANOS
AL.DOS OCENANOS
AL._DOS OCEANOS
ALAM_DOS OCEANOS
ALAM.DOS OCEANOS
ALAM._DOS OCEANOS

Uma vez a porção de texto encontrada, ela pode ser substituída por uma palavra ou pelo resultado de outra expressão regular. Neste caso é feita uma substituição simples da abreviatura pela palavra "ALAMEDA ".

A aplicação executa correções similares para as diversas variantes de abreviaturas de mais de 45 palavras. Estão separadas em dois grupos, que são tratados separadamente e de forma diferente:

- Tipos de via: Rua, Avenida, Praceta, Praça, etc
- Palavras que fazem parte do nome da via: Padre, Professor, Engenheiro, Maria, Janeiro, etc

A etapa seguinte na normalização consiste em uma tarefa mais complexa: a remoção dos elementos desnecessários, porém preservando o número de porta quando este exista. A utilização dos exemplos que seguem visa demonstrar não só as dificuldades encontradas neste ponto, como resumir o raciocínio e processo de desenvolvimento que se teve até o encontro de uma solução que proporcionasse bons resultados. Começando com o exemplo da morada:

```
R REGUEIRA 22 2 DRT
```

Esta morada é um dos casos mais típicos que existem em zonas urbanas. Contém o nome da via, o número de porta, o piso e lado do apartamento. Após as correções de abreviaturas, uma maneira simples de normalizar esta morada seria remover tudo o que estiver a seguir ao primeiro número encontrado, sendo o resultado :

```
RUA REGUEIRA 22
```

Porém, se aplicarmos este mesmo método um outro tipo de morada bastante frequente,

`RUA 25 DE ABRIL 22 2 DRT`

o resultado seria:

`RUA 25`

Um método que funcionaria para processar ambas moradas, seria analisar o texto no sentido inverso, do final para o começo, removendo tudo que exista até o segundo número encontrado.

Temos então

`RUA REGUEIRA 22 2 DRT` que se transforma em `RUA REGUEIRA 22` e

`RUA 25 DE ABRIL 22 2 DRT` que se transforma em `RUA 25 ABRIL 22`.

Entretanto a morada

`R 25 DE ABRIL 22 C/V DRT`, seria modificada para `RUA 25`,

`RUA REGUEIRA LOTE A LOJA 25` seria totalmente eliminada e

`RUA REGUEIRA 22/24/26` se transformaria em `RUA REGUEIRA 22/24`.

As soluções para este tipo de problemas se traduzem em uma serie de expressões regulares complexas executadas de acordo com um fluxo de controle especifico. Os detalhes das soluções encontradas não serão explicitados neste documento por constituírem parte de um produto criado no âmbito empresarial. Mas a titulo ilustrativo da complexidade do processo, segue uma expressão regular que é utilizada em uma das etapas do processo de eliminação de atributos desnecessários:

```
\\W+(\\d+|R/? ?C|S?/?C/?V|esq?(uerd(o|a))?|dto?|d(r|t)o?|frt?e?|fte?|loja)?[^a-z0-
    9\xe3\xf5]{0,3}(esq?(uerd(o|a))?|dto?|d(r|t)o?|frt?e?|fte?|andar|piso)\\.?$
```

O processo de correção e normalização de moradas é extremamente rápido, levando em média 1 segundo para tratar 15 mil registos, deixando-os prontos para a próxima etapa da aplicação: o envio do pedido de *geocoding* para o serviço *Google*.

3.7. Pedido de *geocoding*

3.7.1. Parâmetros do pedido

O único parâmetro obrigatório para a *Geocoding API* retornar um resultado é o *address*. Este parâmetro é a MP para a qual deseja-se saber as coordenadas e pode estar escrito de

diversas formas, da mesma maneira que se escreve na caixa de dialogo da página do *Google Maps* ao procurar um local (Figura 5).

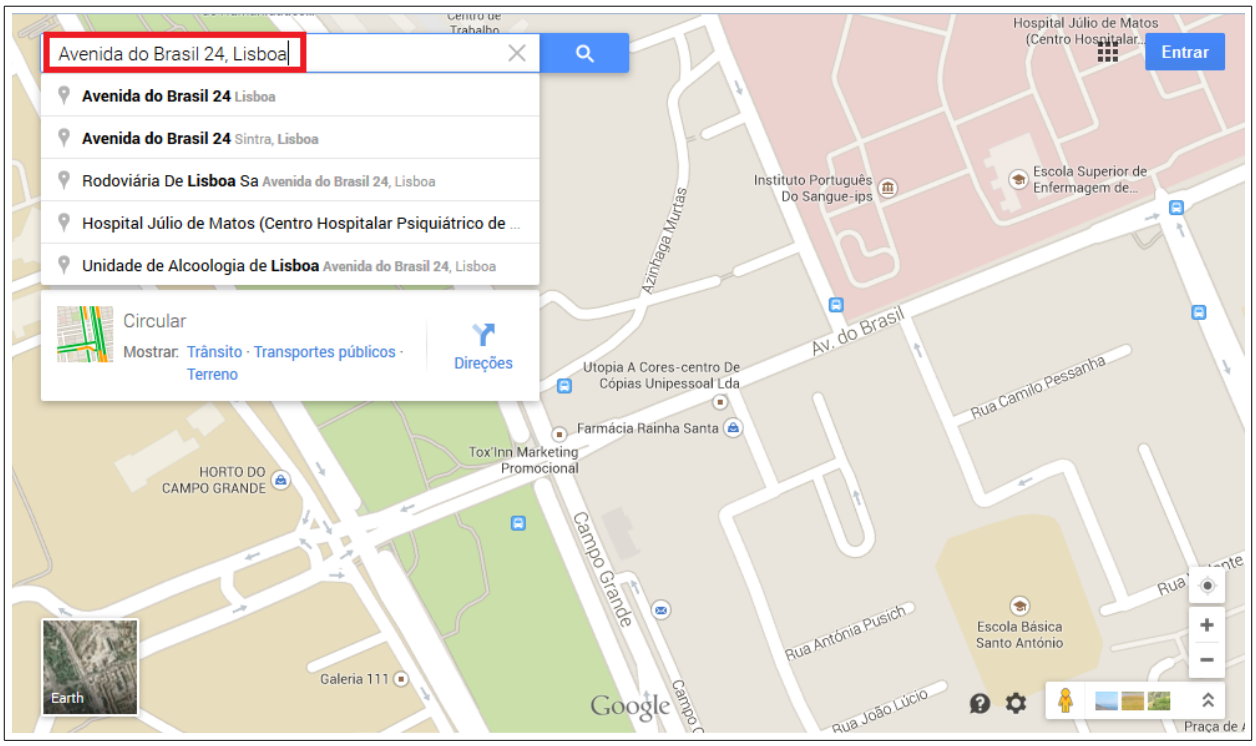


Figura 5: Página do Google Maps com uma pesquisa de morada sendo feita

Os demais parâmetros são opcionais e permitem definir preferências, dar prioridade a determinadas regiões ou filtrar os resultados. O parâmetro ***language*** determina em qual idioma os resultados serão retornados.

Para priorizar uma região na busca da MP, pode-se utilizar o parâmetro ***region***, que recebe como valor um código referente a um país, na forma da sua sigla de domínio de topo (e.g. PT para Portugal ou FR para França). Esta área geográfica será então tida como prioritária na busca, porém os resultados não serão restringidos apenas a esta zona. A *Google* dá o nome a este comportamento na busca de "*biasing*", entretanto a documentação não especifica os detalhes que estão por trás do mesmo.

Pode-se ainda adicionar outro tipo de *biasing* chamado de ***bounds***, dando preferência a busca dentro de uma área retangular definida pelos seus vértices sudoeste e nordeste. A

aplicação utiliza este parâmetro, passando como valor as coordenadas da *bounding-box* do CP4 em questão, extraídas do mapa da *CTT*.

O ultimo parâmetro opcional é o ***filtering***, que diferentemente do ***bounds*** e do ***region***, realmente restringe os resultado a uma localidade ou código postal específico (entre outras possibilidades). Nesta aplicação a utilização deste parâmetro poderia ser prejudicial, pois caso valor do parâmetro utilizado como filtro esteja incorreto (como o concelho ou o código postal por exemplo), seria impossível localizar a MP desejada.

Tomando então como exemplo um pedido para referenciar a MP

`RUA REGUEIRA 22, 3670 CERCOSA, PORTUGAL`,

utilizaríamos os parâmetros:

```
address=RUA REGUEIRA 22, 3670 CERCOSA, PORTUGAL
language=PT
region=PT
bounds=40.60,-8.27|40.75,8.01
```

A URL do pedido seria:

```
maps.googleapis.com/maps/api/geocode/json?address=RUA REGUEIRA 22, 3670 CERCOSA,
PORTUGAL&region=PT&language=PT&bounds=40.60,-8.27|40.75,8.01
```

3.7.2. Criação da assinatura para utilização da *Google Maps API for Work*

A *Google* permite o uso gratuito da *Geocoding API*, porém com um limite de 2500 pedidos diários. Ainda que viável para a georreferenciação de pequenas quantidades de dados, este limite impossibilita o processamento no caso de volumes maiores. Nestas situações deve-se recorrer a API comercial oferecida pela *Google Maps*, com limites muito superiores para todos os serviços. Utilizando a *Google Maps API for Work* o limite passa a ser de cem mil pedidos diários.

Ao utilizar esta *API*, o servidor da *Google* necessita verificar a identidade do cliente que está fazendo o pedido. Para tal, dois parâmetros devem ser adicionados na URL: o ***client_id***, que é o nome de utilizador do cliente, e a ***signature***, uma assinatura que verifica a sua identidade.

A assinatura é gerada utilizando uma chave pessoal e a URL necessária para fazer o pedido, a qual se adiciona o parâmetro ***client_id***. Supondo então um pedido de *geocoding* para Lisboa, a URL para o pedido utilizando a API gratuita seria:

`maps.googleapis.com/maps/api/geocode/json?address=lisboa`

Adicionando o *client_id*, ficaria:

`maps.googleapis.com/maps/api/geocode/json?address=lisboa&client_id=nomeUtilizador`

Extrai-se então a parte que segue o nome de domínio:

`/maps/api/geocode/json?address=lisboa&client_id=nomeUtilizador`

E gera-se a assinatura, aplicando sobre este texto o método de criptografia HMAC-SHA1 (The Internet Engineering Task Force 1997) utilizando a chave pessoal do utilizador. A URL final fica então na forma:

`https://maps.googleapis.com/maps/api/geocode/json?`
`address=lisboa&client_id=nomeUtilizador&signature=assinaturaGerada`

Como foi mencionado anteriormente, existe um módulo em *Python* que facilita o envio de pedidos de *geocoding* para a *Geocoding API* e a recuperação dos resultados. Entretanto, a função deste módulo que tratava de gerar assinaturas para pedidos que utilizassem a *Google Maps API for Work* não estava funcionando corretamente, gerando assinaturas inválidas. Foi nesta parte que contribuiu-se para o projeto *pygeocoder*, corrigindo esta função.

3.7.3. Resposta da *Geocoding API*

Além da coordenada da MP solicitada, a resposta da *Geocoding API* inclui uma série outras informações. Uma delas é o atributo *location_type*, que indica a precisão com que a MP foi referenciada. Este atributo pode ter quatro valores:

Rooftop: O resultado está referenciado com dados discriminados ao número de porta da MP (localização exata na via).

Range Interpolated : A localização da MP ao longo da via foi interpolada a partir de dois pontos com numeração conhecida (a partir do primeiro e ultimo número de um determinado quarteirão, por exemplo)

Geometric Center: A localização corresponde ao centroide de uma via ou região. Entretanto a região pode ser um CP7, um CP4, uma localidade, um concelho ou até mesmo o centroide do país.

Approximate: Indica que a localização é aproximada (sem mais nenhum tipo de especificação).

O grau de confiança dado pelas duas últimas categorias é extremamente variável. Se por um lado o centroide de um CP7 em uma zona urbana pode corresponder a localização de

apenas um ou dois edifícios, por outro, o centroide de um concelho pode abranger centenas de quilómetros quadrados.

Quanto as categorias de melhor precisão (*Rooftop* e *Range Interpolated*), muitas vezes a localização atribuída está completamente errada. Um exemplo comum é quando um pedido de *geocoding* não retorna nenhum resultado, como é o caso para a MP:

ALDEAMENTO TURÍSTICO TORRICENTRO 1, 3808-515 FIGUEIRA DA FOZ, PORTUGAL.

Em uma segunda tentativa de *geocoding*, retira-se a primeira parte da MP, fazendo o pedido com:

3808-515 FIGUEIRA DA FOZ, PORTUGAL

Este pedido retorna uma localização com precisão *Geometric Center* para a MP

RUA FIGUEIRA DA FOZ, 3080 QUIAIOS, PORTUGAL

Claramente não é o resultado correto. Então em uma terceira tentativa, retira-se o nome do concelho, fazendo o pedido unicamente com o CP7 e o país:

3808-515, PORTUGAL

Desta vez o resultado é devolvido com uma precisão do tipo *Rooftop* (localização exata na via), porém para a MP:

PORTUGAL 3808, ITUZAINGÓ, ARGENTINA

Em diversos casos discrepâncias deste tipo foram verificadas. Pedidos de *geocoding* para MPs em Portugal, retornam coordenadas em concelhos errados ou outros países, porém a *Geocoding API* atribui a estes resultados um grau de precisão que indica que a MP está corretamente referenciada. Não se podendo confiar nas precisões atribuídas pelo serviço, foi necessário desenvolver um sistema próprio de atribuição de precisão.

O objeto JSON retornado pela *Geocoding API* conta com atributos do qual se pode tirar partido para isso. De fato, os elementos que constituem a MP retornada têm em geral as suas componentes desagregadas na forma explicitada pela Tabela 3, com os atributos e sua correspondência na divisão territorial Portuguesa.

Tabela 3: Atributos de resposta da Geocoding API

Atributo da resposta da *Geocoding API*	Correspondência no contexto Português
`country`	País
`administrative_area_level_1`	Distrito
`administrative_area_level_2`	Concelho
`administrative_area_level_3`	Freguesia
`locality`	Localidade
`postal_code_prefix`	CP4
`postal_code`	CP7
`route`	Nome da via
`street_number`	Número de porta
`formatted_address`	Morada Postal

Como se conhece a maior parte destes atributos na MP é possível verificar se ela foi bem referenciada, comparando os seus valores com os valores que foram retornados pelo serviço.

3.8. Métricas de precisão dos resultados

A base do sistema de verificação dos resultados é a comparação do valor de atributos da MP original com o valor destes atributos na resposta. Para superar eventuais diferenças na comparação de *strings* que em termos práticos podem corresponder ao mesmo elemento (e.g. OBIDOS e ÓBIDOS), recorreu-se novamente uso da distância de Lenvensthein.

Esta distância corresponde ao número de modificações de caracteres necessárias para que uma *string* fique idêntica a outra. O modulo para *Python fuzzywuzzy* (SeatGeek 2014) baseia-se neste principio para calcular a percentagem de similaridade entre *strings*. Esta percentagem pode ter como referência tanto o tamanho da sequência mais comprida (método `ratio`) como da sequência mais curta (método `partial_ratio`). A segunda opção permite identificar se uma sequência menor existe dentro de uma sequencia maior. Na Tabela 4 vemos alguns exemplos, utilizando ambos métodos:

Com o auxílio desta medida, foi possível comparar cada atributo da resposta com o seu equivalente no pedido. Entretanto, de nada serve saber o grau de similaridade entre dois valores, sem determinar um limiar a partir do qual podem ser considerados idênticos em termos práticos. Um limiar alto garante maior confiança, mas pode deixar de fora muitos valores corretos (falsos-negativos). Utilizar um valor mais baixo reduz o número de falsos negativos,

mas com o risco de considerar como idênticos valores que não o são de fato (falsos positivos). Isto causa uma diminuição da confiança na precisão final atribuída.

Tabela 4: Comparação de strings utilizando o módulo `fuzzywuzzy`

Sequencia 1	Sequencia 2	*ratio*	*partial_ratio*
FIGUEIRA DA FOZ	FIGUEIRA DA FOZ	100	100
CHARNECA DA CAPARICA	CHARNECA DE CAPARICA	95	95
LUZ DE TAVIRA	LUZ (LUZ DE TAVIRA)	81	100
MIRE DE TIBÃES (BRAGA)	MIRE DE TIBAES	70	86
VARZEA DE QUARTEIRA	ESTRADA DE ALBUFEIRA	56	61

De modo empírico, decidiu-se trabalhar com o método ***partial_ratio*** e um limiar de 65% no caso de projetos onde tolera-se uma menor precisão (como para estudos de geo-marketing). Em outras situações, como estudos de otimização de rotas, as localizações devem estar todas corretamente referenciadas, e portanto eleva-se o limiar a 70 ou 75%.

A distância de Levensthein é adequada para comparar sequências de caracteres, mas não é indicada par comparação de números, como são os códigos postais. A similaridade textual entre os CP7 “8100-020” e “1800-200” por exemplo, é de 80%. Entretanto o primeiro se situa em Loulé (Algarve) e o segundo em Lisboa. Para fazer a comparação de códigos postais foi utilizado outro método, verificando quantos dígitos da esquerda para a direita são idênticos entre os dois valores. Com esse sistema, atribuem-se precisões de 0 a 4 para os CP4 e de 0 a 3 para os CP3, de acordo com a Tabela 5.

Tabela 5:Atribuição de precisão para o código postal

Código postal 1	Código postal 2	Precisão CP4	Precisão CP3
1900-223	1900-223	4	3
1900-223	1900-220	4	2
1900-223	1900-200	4	1
1900-223	1900-300	4	0
1900-223	1905-223	3	NA
1900-223	1950-220	2	NA
1900-223	1800-220	1	NA
1900-223	9100-223	0	NA

Uma vez que os CP3 estão incluídos dentro da zona de um CP4, não faz sentido compará-los caso os CP4 não sejam idênticos. Os CP4 tem uma estrutura territorial hierárquica, onde o 1º dígito divide o país em 9 zonas, oito no continente e uma nas ilhas. O segundo dígito corresponde a uma subdivisão destas 9 zonas conforme vemos nos dois mapas da Figura 6. A partir do terceiro dígito já não existe nenhuma hierarquia territorial.

Por este motivo, alguns valores de precisão do código postal tem mais importância que outros. Dada a agrupação hierárquica dos CP4 até ao segundo dígito, assume-se que quando o CP4 da resposta tem uma precisão de 2 ou mais, a zona geográfica onde se encontra a localização retornada pela *Geocoding API* é a correta, permitindo assim que se siga em frente na comparação de atributos com escopo geográfico mais fino, como o nome da via.

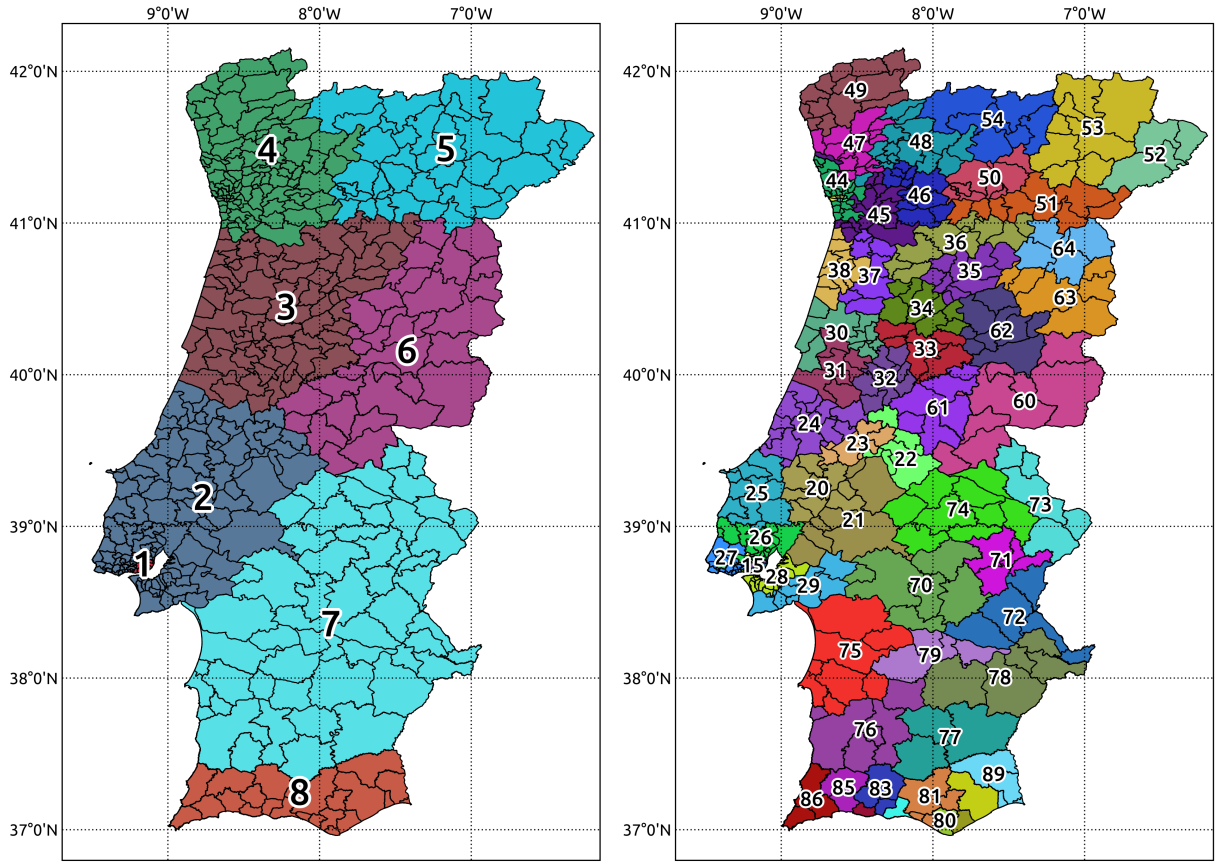


Figura 6: Limites dos CP4 de Portugal continental, agrupados pelos seu primeiro e segundo dígitos

A sucessão de verificações dos atributos é feita de acordo com o fluxograma da Figura 7, atribuindo a cada resultado uma precisão dentre as categorias:

- Mundo: o resultado não se encontra no país correto
- País: o resultado se encontra no país correto
- Concelho: o resultado se encontra no concelho correto
- CP4: o resultado se encontra no CP4 correto
- CP7: o resultado se encontra no CP7 correto. Este grau de precisão é elevado, pois existem mais mais de 30 mil CP7s em Portugal
- Via: o resultado se encontra na via correta
- Porta: o resultado se encontra no ponto correto da via (posição exata ou interpolada)

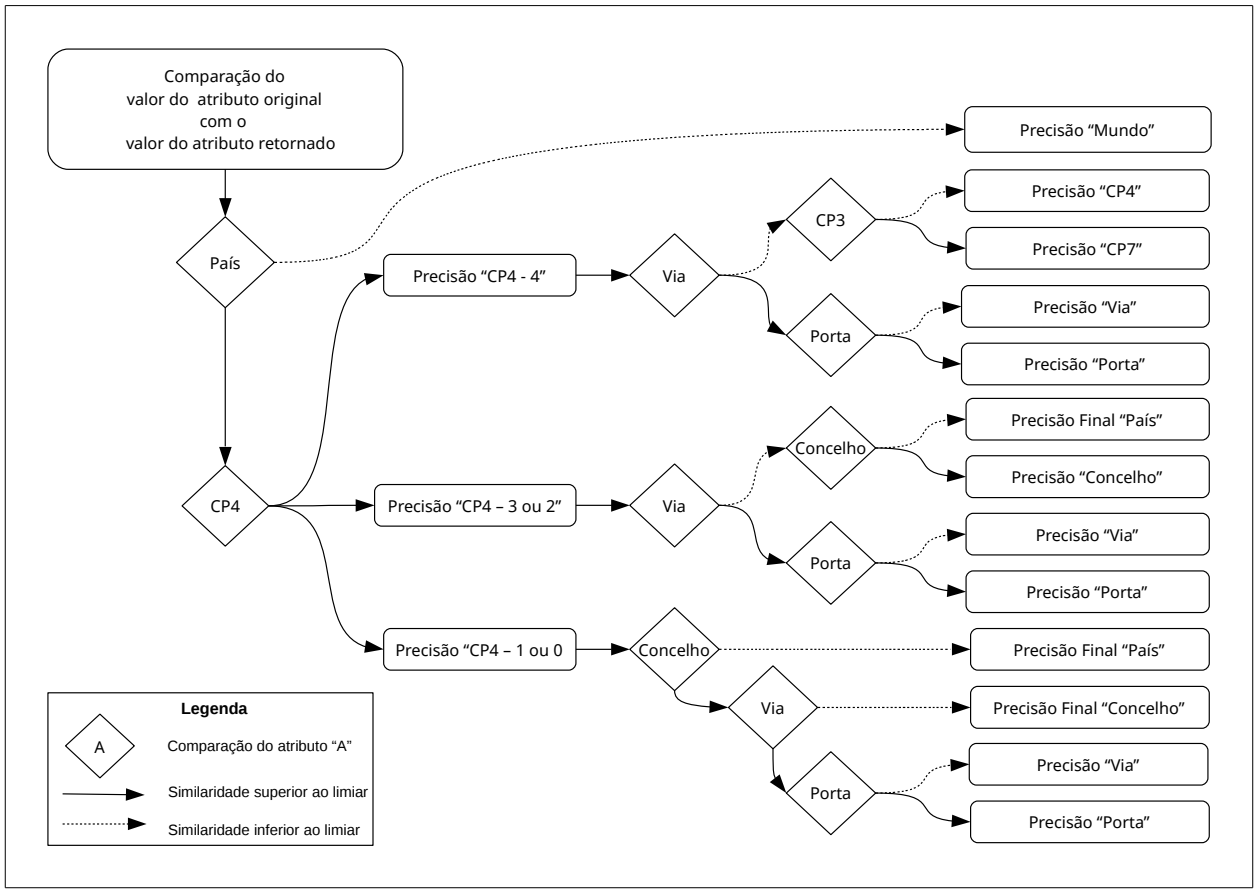


Figura 7: Fluxograma de comparação de atributos para atribuição da precisão ao resultado de geocoding

3.9. Fluxo de controle do g*eocoding*

Foi visto que a composição de uma MP poder ser feita com diferentes elementos de informação, ordenados de várias formas. Também vimos que nem todos os atributos são necessários para o *geocoding*. Idealmente o pedido deve estar no formato:

`Nome da Rua nº, CP4-CP3 Nome do concelho ou localidade, País`

ex: `Avenida de Madrid 2, 1000-096 Lisboa, Portugal`

Porém nem sempre as informações do registo estão todas presente ou corretas, sendo necessário utilizar um formato diferente. A aplicação contempla estes casos definindo funções que criam uma variedade de combinações para uma mesma MP. A escolha do tipo de MP que será utilizado no pedido depende de um fluxo de controle estabelecido na aplicação. Inicia-se o processo com o máximo de informação possível utilizando o formato acima. Avalia-se a precisão da resposta e caso não seja satisfatória elimina-se ou substitui-se gradualmente a informação. O processo segue até que não hajam mais possibilidades, conforme o fluxograma da Figura 8. Neste fluxo de controle, muitas vezes substitui-se o concelho do registo pelo concelho que corresponde ao seu código postal. Esta substituição permite encontrar muitas MPs que não o são na primeira tentativa.

Ao finalizar o processamento dos dados, os resultados são exportados em um ficheiro *CSV* e calculam-se as métricas finais de precisão compostas pelo número e proporção das precisões obtidas.

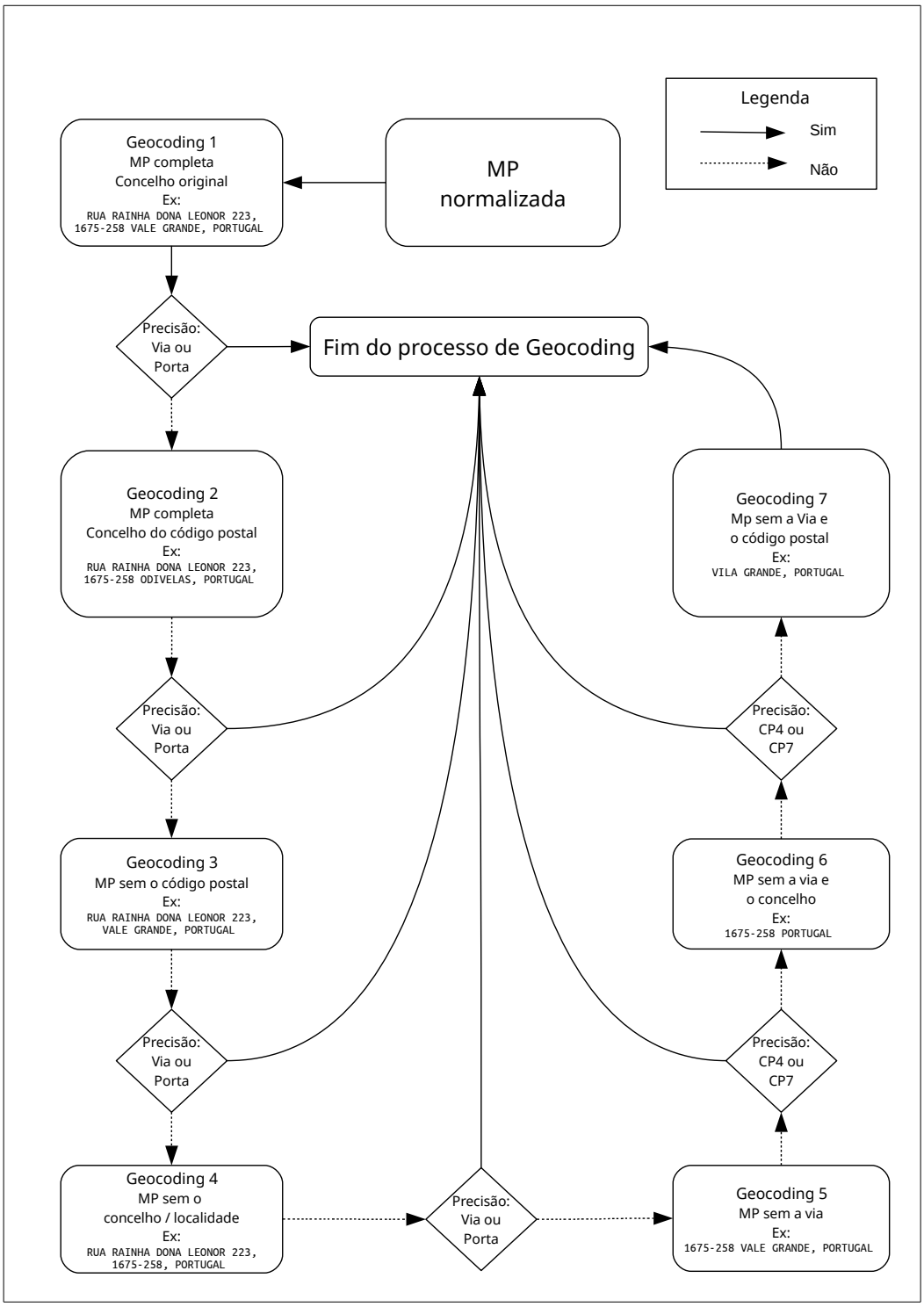


Figura 8: Fluxo de controlo do processo de geocoding

4. RESULTADOS E DISCUSSÃO

4.1. Resultados globais do geocoding e avaliação das métricas de entrada

Ao longo do período de estágio foram referenciados aproximadamente 80 mil registos provenientes de 12 fontes diferentes. Destes, 48 mil foram processados na fase de desenvolvimento da aplicação. Uma vez a mesma considerada estável e funcionando na forma como foi descrita, 31915 registos adicionais, originários de 4 empresas, foram referenciados. É sobre estes dados que se baseiam os resultados a seguir apresentados.

Os resultados globais, tanto da avaliação dos dados de entrada como da precisão da georreferenciação encontram-se discriminados na Tabela 6.

Tabela 6: Resultados das métricas de entrada e precisão do geocoding

		Fonte 1		Fonte 2		Fonte 3		Fonte 4		Total	
		Qtde.	%	Qtde.	%	Qtde.	%	Qtde.	%	Qtde.	%
		1854	100	2196	100	16737	100	11128	100	31915	100
Qualidade dos dados de entrada	CP inexistente	0	0.0%	50	2.3%	5536	33.1%	2	0.0%	5588	17.5%
	CP4 existe	176	9.5%	915	41.7%	395	2.4%	1268	11.4%	2754	8.6%
	CP7 existe	1678	90.5%	1231	56.1%	10806	64.6%	9861	88.6%	23576	73.9%
	Concelho existe	846	45.6%	766	34.9%	6891	41.2%	11131	100.0%	19634	61.5%
	Concelho = CP4	801	43.2%	738	33.6%	4523	27.0%	10977	98.6%	17039	53.4%
Precisão da geo-referenciação	Sem resultados	0	0.0%	1	0.0%	19	0.1%	0	0.0%	20	**0.1%**
	Mundo	0	0.0%	1	0.0%	0	0.0%	0	0.0%	1	**0.0%**
	País	0	0.0%	112	5.1%	1410	8.4%	0	0.0%	1522	**4.8%**
	Concelho	27	1.5%	120	5.5%	1062	6.3%	0	0.0%	1209	**3.8%**
	CP4	270	14.6%	183	8.3%	1172	7.0%	1431	12.9%	3056	**9.6%**
	CP7	236	12.7%	256	11.7%	2454	14.7%	5613	50.4%	8559	**26.8%**
	Via	405	21.8%	425	19.4%	1808	10.8%	1141	10.3%	3779	**11.8%**
	Núm	916	49.4%	1098	50.0%	8812	52.6%	2946	26.5%	13772	**43.2%**

Os registos referenciados com precisão aceitável (“CP7” ou melhor) constituem 81.8% das MPs, sobrando 18.2% dos registos referenciados com precisão entre “CP4” e “Mundo” ou sem um resultado encontrado. A fonte dos dados tem grande influência sobre as precisões obtidas. Para as fontes 1, 2 e 3 foi possível referenciar com precisão alta (“Porta” e “Via”) 71.2%, 69.4% e 63.4% dos dados respetivamente. Na fonte 4, esta proporção é de apenas 36.8%.

Ao comparar-se estes valores com os das métricas de entrada, isto parece não fazer sentido. De acordo com estes, a fonte 4 é a que tem dados de melhor qualidade, logo deveria ter o maior número de registos com precisão alta. É entretanto notável que todos os seus registos ficaram com uma precisão final "CP4" ou melhor e 87.2% com precisão "CP7" ou melhor. Os dados da fonte 1 também têm um comportamento similar: boas métricas de entrada e apenas 1,5% dos registos com precisão inferior a "CP4". Na fonte 3, verifica-se o mesmo, porém no sentido inverso. É a fonte que tem o maior número de códigos postais inexistentes (33.1%) e mais registos com precisão inferior a "CP4" (14.8%).

Conclui-se que as métricas criadas ajudam a prever aproximadamente se os resultados serão referenciados com precisão superior ou inferior ao código postal, mas não são suficientes para previsões mais detalhadas. Isso é normal, pois as métricas avaliam somente o código postal e o concelho dos registos, sendo assim impossível que façam previsões de escopo geográfico mais fino que este.

Para poder antecipar os resultados de forma mais detalhada, deve criar-se uma métrica que trabalhe com o atributo "morada". Um método possível seria buscar moradas cuja estrutura fuja do padrão mais comum. São provavelmente aquelas que não foram capturadas por nenhuma expressão regular e logo não sofreram normalização. Moradas em zonas rurais e industriais também poderiam ser assinaladas pois costumam ser mais difíceis de serem referenciadas.

Apesar de poderem ser melhoradas, as métricas de entrada cumprem sua função, ajustando as expectativas do utilizador quanto à qualidade dos resultados e auxiliando no planeamento do esforço necessário para a referenciação de conjuntos de dados concretos. No âmbito empresarial, tal planeamento é indispensável para a apresentação de propostas orçamentais realistas e fundamentadas.

4.2. Otimização obtida pela normalização e fluxo de controle

A fim de verificar a otimização obtida tanto pela normalização como pelo fluxo de controle dos *geocodings*, foi selecionada uma amostra aleatória composta por 2% dos dados de cada fonte, totalizando 640 registos. Executou-se a georreferenciação destes dados sem qualquer tratamento prévio, apenas uma vez e com todos os atributos disponíveis.

Comparando as precisões obtidas desta forma com as proporcionadas ao processar os mesmos dados na aplicação, as melhorias obtidas podem ser vistas na Tabela 7.

Tabela 7: Precisões obtidas com e sem normalização das moradas

Precisão	Registos originais		Registos normalizados		Diferença	
	Qtde.	%	Qtde.	%		
Sem Resultados	13	2.03%	1	0.16%	-1.88%	-2.19%
Mundo	4	0.63%	2	0.31%	-0.31%	
País	22	3.44%	30	4.69%	1.25%	1.25%
Concelho	42	6.56%	24	3.75%	-2.81%	-8.59%
CP4	112	17.50%	82	12.81%	-4.69%	
CP7	106	16.56%	99	15.47%	-1.09%	
Via	76	11.88%	86	13.44%	1.56%	**9.53%**
Porta	265	41.41%	316	49.38%	7.97%	
Total	640	100.00%	640	100.00%	0.00%	0.00%

Os métodos aqui propostos lograram aumentar o número de registos com precisão "Porta" e "Via" em 19% e 13% respetivamente. Isto se traduz em um aumento de quase 10 pontos percentuais (9.53 p.p.) na quantidade de registos referenciados com precisão alta. Também houve uma diminuição de 2.19 p.p no número de registos não encontrados ou que foram referenciados em um país incorreto.

Estas afirmações tomam como base o sistema de precisões criado e é assim importante verificar qual o nível de confiança do mesmo.

4.3. Avaliação da confiança na atribuição de precisão

Utilizando a mesma amostra aleatória de 640 registos, foi comparada a MP original com a MP devolvida pela *Geocoding API*, atribuindo-se manualmente a precisão correta correspondente. Com as precisões atribuídas manualmente e aquelas calculadas pela pela aplicação, criou-se uma matriz de contingência (Tabela 8) que permite avaliar a qualidade do processo. A precisão global obtida foi de 91,72%.

Tabela 8: Matriz de contingência da avaliação manual de precisões

	Atribuição automática de precisão									
Atribuição manual de precisão	**Porta**	**Via**	**CP7**	**CP4**	**Concelho**	**Pais**	**Mundo**	**Sem resultado**	**Total**	**Precisão do utilizador**
Porta	**291**	1	0	1	0	0	0	0	293	99.32%
Via	6	**77**	1	1	0	0	0	0	85	90.59%
CP7	1	2	**98**	0	0	0	0	0	101	97.03%
CP4	9	5	0	**80**	0	0	0	0	94	85.11%
Concelho	1	1	0	0	**24**	16	0	0	42	57.14%
Pais	8	0	0	0	0	**14**	0	0	22	63.64%
Mundo	0	0	0	0	0	0	**2**	0	2	100.00%
Sem resultado	0	0	0	0	0	0	0	**1**	1	100.00%
Total	316	86	99	82	24	30	2	1	640	
Precisão do produtor	92.09%	89.53%	98.99%	97.56%	100.00%	46.67%	100.00%	100.00%		

Do ponto de vista do produtor, a única categoria mal classificada é a "País". Mais da metade dos registos desta classe deveriam estar com precisão "Concelho". O nome dos concelhos são *strings* pequenas, fazendo com que pequenas diferenças entre dois valores no momento da medição da distância de Levensthein baixem muito a percentagem de similaridade. Caso esta fique abaixo do limiar estabelecido (65%), um registo com o concelho correto acaba por ser classificado com precisão "País". Um exemplo é a comparação das strings "Algueirão - Mem Martinz" (concelho do pedido) e "Algueirão" (concelho da resposta), com 60% de similaridade, fazendo com que o registo seja classificado com precisão "País", apesar da resposta se encontrar no concelho correto.

Outras vezes os erros de classificação se devem à falta de dados na resposta. A coordenada retornada está correta porém grande parte dos elementos que a constituem não estão presentes. Por exemplo, um pedido para a MP:

```
RUA ANÇARIZ, 4770-360 VILA NOVA DE FAMALICÃO, PORTUGAL
```

além das coordenadas, retorna apenas os valores:

```
formatted_address:              Rua de Ançariz, Portugal
country:                        Portugal
administrative_area_level_1:    Braga
```

Somente com estes dados não é possível avaliar de forma automática a precisão deste resultado. Uma solução que permitiria ultrapassar esta dificuldade seria utilizar as coordenadas devolvidas pela *Geocoding API* para identificar na Carta Administrativa Oficial de Portugal

(CAOP) (Direção-Geral do Território 2014) qual o Concelho e Freguesia referentes à aquela localização.

Do ponto de vista do utilizador, este problema traduz-se em uma baixa precisão das classes "País" e "Concelho". Mas em todo caso, o método proposto de avaliação dos resultados contém mais detalhe do que aquele oferecido pela *Geocoding API*. Oferece 7 classes de precisão que são baseadas em amplitude espacial, enquanto que o serviço da *Google* oferece apenas 4, baseadas nos dados de origem e na forma como a localização foi obtida. A confiança da avaliação oferecida pela aplicação também supera a da *Geocoding API*, como será visto adiante.

4.4. Comparação das precisões atribuídas pela *Geocoding API* com as atribuídas pela aplicação

Supondo que as precisões atribuídas pela *Geocoding API* estejam sempre corretas, a correspondência entre os seus valores e aqueles determinados pela aplicação teria a estrutura da Tabela 9:

Tabela 9: Correspondência das precisões da Geocoding API e da aplicação

	Precisão da aplicação							
Precisão da *Geocoding API*	Numero	Via	CP7	CP4	Concelho	País	Mundo	Sem Resultados
Rooftop	x							
Range Interpolated	x							
Geometric Center		x	x	x	x	x	x	
Approximate		x	x	x	x	x	x	
Zero Results								x

As precisões da *Geocoding API Geometric Center* e *Approximate* podem corresponder a 6 níveis de precisão diferentes na aplicação. Assim, a única forma possível de comparar os dois métodos é separando os registos em duas categorias: aqueles com precisão "Porta" e aqueles com precisão inferior a "Porta" ("Inf. Porta"). A semelhança da análise prévia foram feitas mais duas tabelas de contingência, de acordo com as duas novas categorias estabelecidas. A Tabela 10 compara os resultados da verificação manual com os fornecidos pela API. A Tabela 11 compara os da verificação manual com os calculados pela aplicação.

Tabela 10: Matriz de contingências das precisões verificadas manualmente e segundo a Geocoding API

	Precisão da *Geocoding API*			
Precisão verificada manualmente	Porta	Inf. Porta	Total	Precisão do Utilizador
Porta	293	0	293	100.00 %
Inf. Porta	52	295	347	85.01 %
Total	345	295	640	
Precisão do produtor	84.93 %	100.00 %		

Tabela 11: Matriz de contingências das precisões atribuídas pela aplicação e verificadas manualmente

	Precisão da aplicação			
Precisão verificada manualmente	Porta	Inf. Porta	Total	Precisão do Utilizador
Porta	291	2	293	99.32 %
Inf. Porta	25	322	347	92.80 %
Total	316	324	640	
Precisão do produtor	92.09 %	99.38 %		

Para a classe "Porta", a precisão do produtor no caso da *Geocoding API* é de 84.93%, inferior a da aplicação, que é de 92.09 %. Para a classe "Inf. Porta", a precisão é de 100% para a API e de 99.38% para a aplicação. A aplicação tem neste caso uma precisão ligeiramente inferior a *Geocoding API*, porém deve-se lembrar que a classe "Inf. Porta" corresponde a apenas 2 classes na *Geocoding API*, e a 6 classes na aplicação.

Tendo apenas duas classes nesta análise, as precisões do Utilizador seguem a mesma dinâmica, porem com as categorias invertidas. Conclui-se que o método apresentado não só oferece mais detalhe no nível de precisão dos resultados, como maior confiança na sua classificação.

5. CONSIDERAÇÃOS FINAIS

A aplicação desenvolvida deu origem a uma ferramenta que possibilita a georreferenciação massiva de dados de forma eficaz. O *geocoding* é um processo que dificilmente será algum dia isento de erros e a precisão final obtida pelos métodos aqui apresentados é muito satisfatória (81.8 % dos dados com precisão entre "CP7" e "Porta") dentro do seu contexto de utilização.

A normalização dos dados e o sistema de múltiplos *geocodings* para uma mesma MP claramente aumentaram a qualidade dos resultados. Porém a etapa de normalização sempre será passível de melhorias. Conforme mais dados vão sendo recebidos e processados, é importante analisar onde os resultados não foram satisfatórios, para que as expressões regulares contemplem cada vez mais abreviaturas e estruturas de moradas não comuns. Também é de se esperar que a completude da base de dados do *Google Maps* aumente com o tempo, assim como a eficácia de seu algoritmo de *geocoding*.

Apesar de oferecer bons resultados na georreferenciação, as precisões atribuídas pela *Geocoding API* não são detalhadas o suficiente no âmbito das atividades da *Focus BC*. Um método de compreensão mais palpável foi assim criado e sua confiança é maior que aquela do serviço.

Este método oferece boa confiança, mas não é infalível. Os erros que são consequência do uso da distância de Levensthein podem eventualmente serem reduzidos com a utilização de outros algoritmos e métodos para comparação de *strings*. Existem várias abordagens possíveis para a resolução de problemas deste tipo, denominados de *Approximate String Mathcing* e que também são enfrentados nas áreas de correção ortográfica, sequenciamento de ADN e processamento de sinais (Navarro 2001). Um estudo sucinto das técnicas disponíveis deve ser feito para escolher a mais adequada. Uma forma mais simples de melhorar a confiança das precisões é incorporar a CAOP no método, obtendo assim mais informação passível de comparação. Esta funcionalidade encontra-se em fase de implementação.

A aplicação foi criada no contexto de Portugal, o que implica que está adaptada ao seu idioma, estrutura das MPs e organização territorial. Tal qual se encontra agora, não é possível utiliza-la com MPs estrangeiras, porém os métodos e fluxos de controle criados podem ser facilmente adaptados para outros países. As divisões territoriais e estrutura das MPs são muito similares entre os países ocidentais. (Li & Zhang 2010).

A parte mais trabalhosa da adaptação seria na parte de normalização das moradas pois todas as expressões regulares devem ser refeitas de acordo com o novo idioma. Para adaptação do fluxo de *geocoding* e atribuição de precisão, apenas é necessário analisar a hierarquia territorial do país em questão, e substituir os elementos correspondentes, conforme os exemplos da Tabela 12.

Tabela 12: Atributos de resposta do geocoding e correspondência territorial em Portugal, E.U.A. e França

Atributo da resposta de *geocoding*	Correspondência no contexto Português	Correspondência no contexto Americano	Correspondência no contexto Francês
`country`	País	País	País
`administrative_area_level_1`	Distrito	Estado	Região
`administrative_area_level_2`	Concelho	Condado	Departamento
`administrative_area_level_3`	Freguesia	N/A	*Arrondissement*
`locality`	Localidade	Cidade	Comuna
`postal_code_prefix`	CP4	ZIP (5 dígitos)	N/A
`postal_code`	CP7	ZIP+4 (9 dígitos)	CP de 5 dígitos
`route`	Nome da via	Nome da via	Nome da Via
`street_number`	Número de porta	Número de porta	Número de porta

Outro caminho interessante para ser explorado é a criação de uma interface gráfica que permita a qualquer utilizador beneficiar das otimizações que a aplicação faz sobre a *Geocoding API*. A interface encontra-se em fase de desenvolvimento, na forma de uma aplicação web suportada pela *framework Django* (Django Software Foundation 2014) que também é escrita em *Python*, permitindo que se reutilize grande parte do código já escrito.

BIBLIOGRAFIA

Bittner, T. & Stell, J.G., 1999. A Boundary-Sensitive Approach to Qualitative Location. *Annals of Mathematics and Artificial Intelligence*, pp.1–24.

Cote, P., 2014. GIS Manual: Spatial Information in Design Culture. *Harvard University - Graduate School of Design*. Available at: http://www.gsd.harvard.edu/gis/manual/projection_fundamentals/index.htm [Accessed September 4, 2014].

CTT - Correios de Portugal SA, 2014. CTT - Ferramentas. Available at: www.ctt.pt/feapl_2/app/open/tools.jspx?tool=1 [Accessed September 12, 2014].

Davis Jr., C.A. & Fonseca, F.T., 2007. Assessing the Certainty of Locations Produced by an Address Geocoding System. *GeoInformatica*, 11, pp.103–129.

Direção-Geral do Território, 2014. CAOP 2014. Available at: http://www.dgterritorio.pt/cartografia_e_geodesia/cartografia/carta_administrativa_oficial_de_portugal__caop_/caop_em_vigor/ [Accessed September 17, 2014].

Django Software Foundation, 2014. Django. Available at: https://www.djangoproject.com/ [Accessed September 20, 2014].

ECMA International, 2011. EcmaScript Language Specification. Available at: http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf.

ECMA International, 2013. The JSON Data Interchange Format. , (October). Available at: http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf.

ESRI, 2014. ArcGIS for Desktop. Available at: http://www.esri.com/software/arcgis/arcgis-for-desktop.

Felice, G. De, 2012. *Reasoning with mixed qualitative-quantitative representtations of spatial knowledge*. Universität Bremen.

Fielding, R.T., 2000. *Architectural Styles and the Design of Network-based Software Architectures*. University of California.

Focus BC, 2014. Focus BC. Available at: http://www.focus-bc.com/pt/ [Accessed September 17, 2014].

Friedl, J.E.F., 2002. *Mastering regular expressions* 2nd ed. A. Oram, ed., Sebastopol: O'Reilly & Associates, Inc.

Geographical Publishing Co., 1905. Premier series map of Oklahoma and Indian Territory. *Daily Oklahoman.* Available at: http://hdl.loc.gov/loc.gmd/g4020.ct000282 .

Goldberg, D.W., 2008. *A Geocoding Best Practices Guide*, Springfield: North American Association of Central Cancer Registries, Inc.

Google, 2014a. Google Earth Pro. Available at: https://www.google.com/enterprise/mapsearth/products/earthpro.html [Accessed September 17, 2014].

Google, 2014b. Google Maps. Available at: https://www.google.com/maps/about/ [Accessed September 10, 2014].

Google, 2014c. Google Maps API Web Services. Available at: https://developers.google.com/maps/documentation/webservices/ [Accessed September 12, 2014].

Google, 2014d. Google Maps Coordinate. Available at: https://www.google.com/enterprise/mapsearth/products/coordinate.html [Accessed September 14, 2014].

Google, 2014e. Google Maps Engine. Available at: https://www.google.com/enterprise/mapsearth/products/mapsengine.html [Accessed September 17, 2014].

Google, 2014f. Google Maps Javascript API V3 Reference. Available at: https://developers.google.com/maps/documentation/javascript/reference [Accessed September 12, 2014].

Google, 2014g. The Google Elevation API. Available at: https://developers.google.com/maps/documentation/elevation/ [Accessed September 12, 2014].

Gosling, J. & McGilton, H., 1996. The Java Language Environment. Available at: http://www.oracle.com/technetwork/java/langenv-140151.html [Accessed September 12, 2014].

Hill, L., 2006. *Georeferencing: The Geographic Associations of Information.*, The MIT Press.

incratec GmbH, 2014. GeoCoder Pro. Available at: http://www.geocoderpro.com/ [Accessed September 10, 2014].

Levensthein, V., 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8), pp.707–710. Available at: http://profs.sci.univr.it/~liptak/ALBioinfo/files/levenshtein66.pdf.

Li, B. & Zhang, X., 2010. Automatic construction and visualization of address models. *2010 Sixth International Conference on Natural Computation*, (Icnc), pp.2894–2897. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5584218.

Microsoft, 2014. Bing Maps. Available at: http://www.microsoft.com/maps/ [Accessed September 10, 2014].

Navarro, G., 2001. A guided tour to approximate string matching. *ACM Computing Surveys*, 33, pp.31–88. Available at: http://portal.acm.org/citation.cfm?doid=375360.375365.

Navteq Maps, 2014. Navstreets Digital Street Network. Available at: http://www.navmart.com/navteq_navstreets.php [Accessed September 9, 2014].

Numpy developers, 2014. NumPy. Available at: http://www.numpy.org/ [Accessed September 12, 2014].

OpenStreetMap Contributors, 2014. OpenSteetMap. Available at: http://www.openstreetmap.org/about [Accessed September 9, 2014].

Oracle, 2014. Oracle Spatial and Graph. Available at: http://www.oracle.com/technetwork/database/options/spatialandgraph/overview/spatialandgraph-1707409.html.

PostGIS Project Steering Committee, 2014. PostGIS. Available at: http://postgis.net/ [Accessed September 12, 2014].

Python Software Foundation, 2014a. About Python. Available at: https://www.python.org/about/ [Accessed September 12, 2014].

Python Software Foundation, 2014b. Regular expression operations. *The Python Standard Library*. Available at: https://docs.python.org/2/library/re.html [Accessed September 12, 2014].

SeatGeek, 2014. Fuzzy String Matching in Python. Available at: https://github.com/seatgeek/fuzzywuzzy [Accessed September 12, 2014].

The Internet Engineering Task Force, 2005. Common Format and MIME Type for Comma-Separated Values (CSV) Files. Available at: http://tools.ietf.org/html/rfc4180 [Accessed September 12, 2014].

The Internet Engineering Task Force, 1997. HMAC: Keyed-Hashing for Message Authentication. , p.11. Available at: http://tools.ietf.org/pdf/rfc2104.pdf.

TomTom, 2014. Multinet. Available at: http://www.tomtom.com/en_gb/licensing/products/maps/multinet/ [Accessed September 9, 2014].

U.S. Census Bureau, 2014. TIGER Products. Available at: https://www.census.gov/geo/maps-data/data/tiger.html [Accessed September 9, 2014].

United Nations Group of Experts on Geographical Names, 2006. *Manual for the national standardization of geographical names*, New York.

Universal Postal Union, 2013. Postal addressing systems in member countries - Portugal. Available at: http://www.upu.int/fileadmin/documentsFiles/activities/addressingUnit/prtEn.pdf.

World Wide Web Consortium, 2008. Extensible Markup Language (XML) 1.0 (Fifth Edition). Available at: http://www.w3.org/TR/xml/ [Accessed September 12, 2014].

Yao, X. & Jiang, B., 2005. Visualization of Qualitative Locations in Geographic Information Systems. *Cartography and Geographic Information Science*, 32(4), pp.219–229. Available at: http://www.tandfonline.com/doi/abs/10.1559/152304005775194683.

Yu, X., 2014. pygeocoder. Available at: https://bitbucket.org/xster/pygeocoder/wiki/Home [Accessed September 12, 2014].